

# 5243\_proj\_5

December 12, 2023

In this project, we delved into the analysis of..

## 0.0.1 Data Processing

The data file imdb\_reviews\_unique.csv is scraped from imdb with columns...

```
[ ]: import pandas as pd
from transformers import pipeline
import nltk
nltk.download('punkt')
from nltk.tokenize import sent_tokenize

file_path = './imdb_reviews_unique.csv'
reviews_df = pd.read_csv(file_path)

sentiment_model = pipeline("sentiment-analysis", truncation=True)

def analyze_sentiment(title, review):
    full_text = title + ". " + review
    sentences = sent_tokenize(full_text)
    scores = []

    for sentence in sentences:
        max_length = 512
        parts = [sentence[i:i + max_length] for i in range(0, len(sentence),
↪max_length)]

        for part in parts:
            if part.strip() != '':
                result = sentiment_model(part)[0]
                scores.append(result['score'] if result['label'] == 'POSITIVE'
↪else 1 - result['score'])

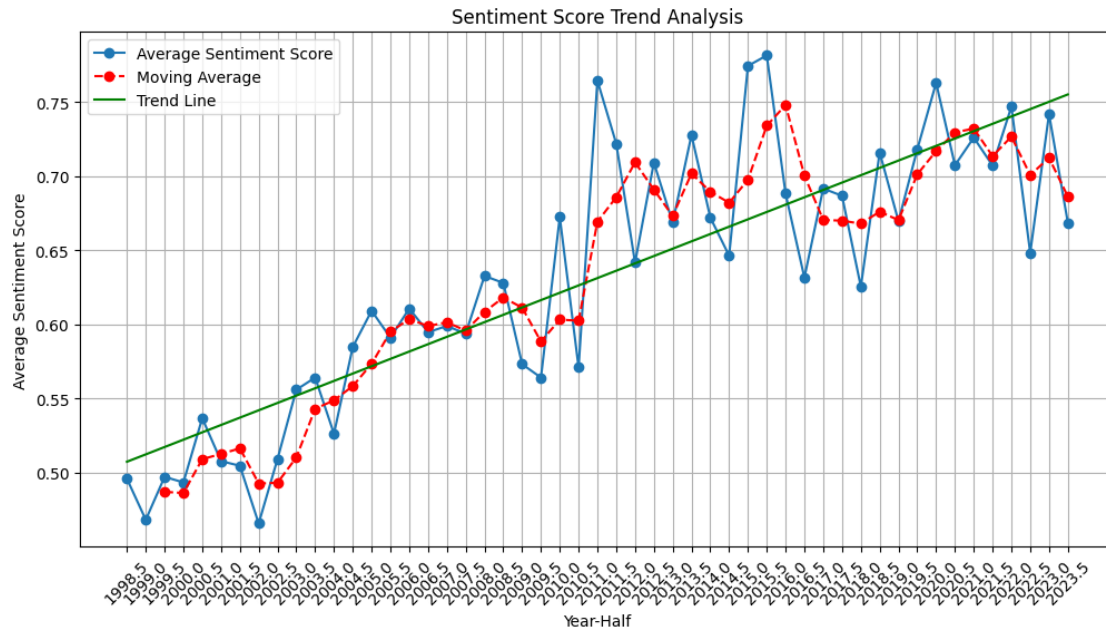
    return sum(scores) / len(scores) if scores else 0

reviews_df['Sentiment_Score'] = reviews_df.apply(lambda x:
↪analyze_sentiment(x['Title'], x['Review']), axis=1)
```

```
reviews_df.head()
```

```
[ ]: output_path = "/content/Sentiment_Score.csv"  
reviews_df.to_csv(output_path, index=False)
```

```
[ ]: import pandas as pd  
import numpy as np  
from sklearn.linear_model import LinearRegression  
import matplotlib.pyplot as plt  
  
reviews_df['Date'] = pd.to_datetime(reviews_df['Date'])  
  
reviews_df['Year-Half_Numeric'] = reviews_df['Date'].apply(lambda x: x.year +  
↳ (0.5 if x.month > 6 else 0))  
  
score_by_time = reviews_df.groupby('Year-Half_Numeric')['Sentiment_Score'].  
↳ mean().reset_index()  
  
score_by_time['Moving_Avg'] = score_by_time['Sentiment_Score'].  
↳ rolling(window=3).mean()  
  
X = score_by_time['Year-Half_Numeric'].values.reshape(-1, 1)  
y = score_by_time['Sentiment_Score'].values  
  
model = LinearRegression().fit(X, y)  
score_by_time['Trend'] = model.predict(X)  
  
plt.figure(figsize=(12, 6))  
plt.plot(score_by_time['Year-Half_Numeric'], score_by_time['Sentiment_Score'],  
↳ marker='o', label='Average Sentiment Score')  
plt.plot(score_by_time['Year-Half_Numeric'], score_by_time['Moving_Avg'],  
↳ marker='o', linestyle='--', color='red', label='Moving Average')  
plt.plot(score_by_time['Year-Half_Numeric'], score_by_time['Trend'],  
↳ linestyle='-', color='green', label='Trend Line')  
  
plt.xticks(score_by_time['Year-Half_Numeric'], rotation=45)  
plt.xlabel('Year-Half')  
plt.ylabel('Average Sentiment Score')  
plt.title('Sentiment Score Trend Analysis')  
plt.legend()  
plt.grid(True)  
plt.show()
```



```
[ ]: import pandas as pd
import scipy.stats as stats
import matplotlib.pyplot as plt
import seaborn as sns

reviews_df_path = '/content/Sentiment_Score.csv'
imdb_reviews_path = '/content/imdb_reviews_rating.csv'

reviews_df = pd.read_csv(reviews_df_path)
imdb_reviews = pd.read_csv(imdb_reviews_path)

merged_df = pd.merge(reviews_df, imdb_reviews[['Author', 'Rating']],
    on='Author', how='left')

merged_df = merged_df[merged_df['Rating'] != 'No Rating']
merged_df.dropna(subset=['Rating'], inplace=True)

merged_df['Rating'] = merged_df['Rating'].str.split('/').str[0]
merged_df['Rating'] = pd.to_numeric(merged_df['Rating'], errors='coerce')

merged_df['Transformed_Sentiment_Score'] = (merged_df['Sentiment_Score'] * 9)+1

transformed_correlation = merged_df[['Transformed_Sentiment_Score', 'Rating']].
    corr()
```

```

spearman_corr_transformed, spearman_p_value_transformed = stats.
↳spearmanr(merged_df['Transformed_Sentiment_Score'], merged_df['Rating'],
↳nan_policy='omit')

print("Correlation Matrix:\n", transformed_correlation)
print("\nSpearman Correlation Coefficient: ", spearman_corr_transformed)
print("Spearman P-value: ", spearman_p_value_transformed)

print("\nDescriptive Statistics:\n", merged_df[['Transformed_Sentiment_Score',
↳'Rating']].describe())

plt.figure(figsize=(12, 6))

plt.subplot(1, 2, 1)
sns.boxplot(data=merged_df, y='Transformed_Sentiment_Score')
plt.title('Boxplot of Transformed Sentiment Score')

plt.subplot(1, 2, 2)
sns.boxplot(data=merged_df, y='Rating')
plt.title('Boxplot of Rating')

plt.tight_layout()
plt.show()

```

Correlation Matrix:

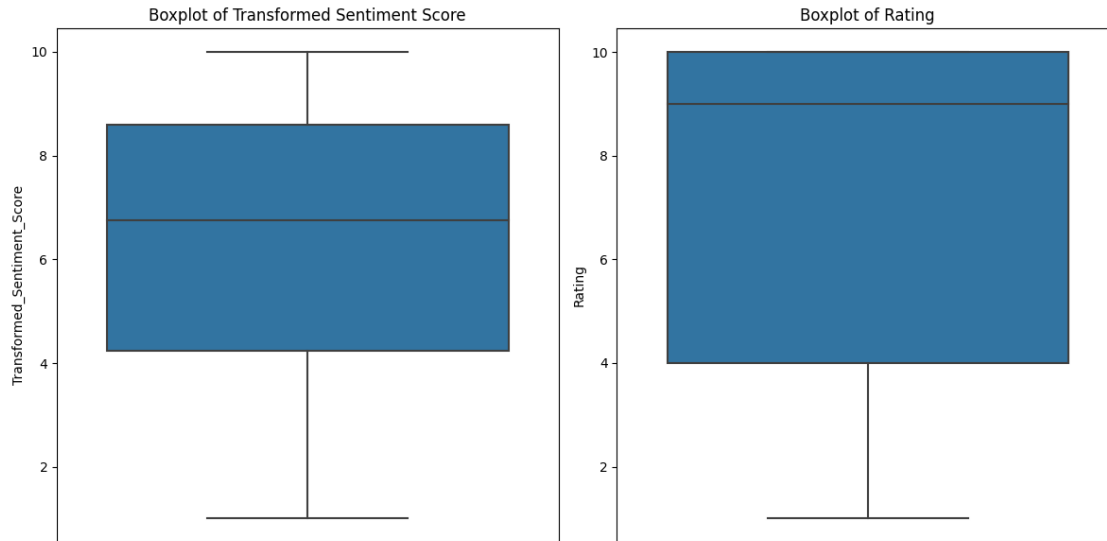
	Transformed_Sentiment_Score	Rating
Transformed_Sentiment_Score	1.000000	0.757396
Rating	0.757396	1.000000

Spearman Correlation Coefficient: 0.7055647694706387

Spearman P-value: 2.1264551409578257e-181

Descriptive Statistics:

	Transformed_Sentiment_Score	Rating
count	1200.000000	1200.000000
mean	6.383814	7.029167
std	2.624475	3.484983
min	1.002166	1.000000
25%	4.233460	4.000000
50%	6.760628	9.000000
75%	8.589163	10.000000
max	9.998989	10.000000



# 1 Word Frequency

In this section, we are working on ...

```
[ ]:
```

```
[ ]:                                     Title      Author \
0                                     bravo      Rage-4
1 How many times I watch this movie... It's sti... TaylorYee94
2          STANDING OVATION! - (10 stars out of 10)  BJG-Reviews
3 Despite a lot of plot flaws and conveniences, ... paulclaassen
4          Great      MrHeraclius

                                Review  Sentiment_Score  Year
0 Firstly and foremostly, I am a guy, which (in ...      0.638083  1970
1 I have watched Titanic how many times I don't ...      0.758639  1970
2 The stage curtains open ...Not since the adven...      0.810825  1970
3 Ah, yes, the film that propelled Leonardi DiCa...      0.775845  1970
4 Very beautiful and cinematic movie with lots o...      0.999344  1970
```

```
[ ]: !pip install spacy==2.3.5
      !python -m spacy download en_core_web_sm
```

```
Collecting spacy==2.3.5
  Downloading spacy-2.3.5.tar.gz (5.8 MB)
                                5.8/5.8 MB
19.4 MB/s eta 0:00:00
  Installing build dependencies ... done
```

```

Getting requirements to build wheel ... done
Installing backend dependencies ... done
Preparing metadata (pyproject.toml) ... done
Requirement already satisfied: murmurhash<1.1.0,>=0.28.0 in
/usr/local/lib/python3.10/dist-packages (from spacy==2.3.5) (1.0.10)
Requirement already satisfied: cymem<2.1.0,>=2.0.2 in
/usr/local/lib/python3.10/dist-packages (from spacy==2.3.5) (2.0.8)
Requirement already satisfied: preshed<3.1.0,>=3.0.2 in
/usr/local/lib/python3.10/dist-packages (from spacy==2.3.5) (3.0.9)
Collecting thinc<7.5.0,>=7.4.1 (from spacy==2.3.5)
Using cached
thinc-7.4.6-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (1.0 MB)
Requirement already satisfied: blis<0.8.0,>=0.4.0 in
/usr/local/lib/python3.10/dist-packages (from spacy==2.3.5) (0.7.11)
Collecting wasabi<1.1.0,>=0.4.0 (from spacy==2.3.5)
Using cached wasabi-0.10.1-py3-none-any.whl (26 kB)
Collecting srsly<1.1.0,>=1.0.2 (from spacy==2.3.5)
Using cached
srsly-1.0.7-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (369 kB)
Collecting catalogue<1.1.0,>=0.0.7 (from spacy==2.3.5)
Using cached catalogue-1.0.2-py2.py3-none-any.whl (16 kB)
Requirement already satisfied: tqdm<5.0.0,>=4.38.0 in
/usr/local/lib/python3.10/dist-packages (from spacy==2.3.5) (4.66.1)
Requirement already satisfied: setuptools in /usr/local/lib/python3.10/dist-
packages (from spacy==2.3.5) (67.7.2)
Requirement already satisfied: numpy>=1.15.0 in /usr/local/lib/python3.10/dist-
packages (from spacy==2.3.5) (1.23.5)
Collecting plac<1.2.0,>=0.9.6 (from spacy==2.3.5)
Using cached plac-1.1.3-py2.py3-none-any.whl (20 kB)
Requirement already satisfied: requests<3.0.0,>=2.13.0 in
/usr/local/lib/python3.10/dist-packages (from spacy==2.3.5) (2.31.0)
Requirement already satisfied: charset-normalizer<4,>=2 in
/usr/local/lib/python3.10/dist-packages (from
requests<3.0.0,>=2.13.0->spacy==2.3.5) (3.3.2)
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-
packages (from requests<3.0.0,>=2.13.0->spacy==2.3.5) (3.6)
Requirement already satisfied: urllib3<3,>=1.21.1 in
/usr/local/lib/python3.10/dist-packages (from
requests<3.0.0,>=2.13.0->spacy==2.3.5) (2.0.7)
Requirement already satisfied: certifi>=2017.4.17 in
/usr/local/lib/python3.10/dist-packages (from
requests<3.0.0,>=2.13.0->spacy==2.3.5) (2023.11.17)
Building wheels for collected packages: spacy
error: subprocess-exited-with-error

```

× Building wheel for spacy

(pyproject.toml) did not run successfully.

```
exit code: 1
> See above for output.
```

**note:** This error originates from a subprocess, and is likely not a problem with pip.

Building wheel for spacy (pyproject.toml) ... error

**ERROR: Failed building wheel for spacy**

Failed to build spacy

**ERROR: Could not build wheels for spacy, which is required to install**

**pyproject.toml-based projects**

```
2023-12-12 06:12:31.010144: E
tensorflow/compiler/xla/stream_executor/cuda/cuda_dnn.cc:9342] Unable to
register cuDNN factory: Attempting to register factory for plugin cuDNN when one
has already been registered
```

```
2023-12-12 06:12:31.010237: E
tensorflow/compiler/xla/stream_executor/cuda/cuda_fft.cc:609] Unable to register
cuFFT factory: Attempting to register factory for plugin cuFFT when one has
already been registered
```

```
2023-12-12 06:12:31.010296: E
tensorflow/compiler/xla/stream_executor/cuda/cuda_blas.cc:1518] Unable to
register cuBLAS factory: Attempting to register factory for plugin cuBLAS when
one has already been registered
```

```
2023-12-12 06:12:33.271069: W
tensorflow/compiler/tf2tensorrt/utils/py_utils.cc:38] TF-TRT Warning: Could not
find TensorRT
```

Collecting en-core-web-sm==3.6.0

Downloading [https://github.com/explosion/spacy-](https://github.com/explosion/spacy-models/releases/download/en_core_web_sm-3.6.0/en_core_web_sm-3.6.0-py3-none-any.whl)  
models/releases/download/en\_core\_web\_sm-3.6.0/en\_core\_web\_sm-3.6.0-py3-none-  
any.whl (12.8 MB)

12.8/12.8 MB

27.0 MB/s eta 0:00:00

Requirement already satisfied: spacy<3.7.0,>=3.6.0 in  
/usr/local/lib/python3.10/dist-packages (from en-core-web-sm==3.6.0) (3.6.1)

Requirement already satisfied: spacy-legacy<3.1.0,>=3.0.11 in  
/usr/local/lib/python3.10/dist-packages (from spacy<3.7.0,>=3.6.0->en-core-web-  
sm==3.6.0) (3.0.12)

Requirement already satisfied: spacy-loggers<2.0.0,>=1.0.0 in  
/usr/local/lib/python3.10/dist-packages (from spacy<3.7.0,>=3.6.0->en-core-web-  
sm==3.6.0) (1.0.5)

Requirement already satisfied: murmurhash<1.1.0,>=0.28.0 in  
/usr/local/lib/python3.10/dist-packages (from spacy<3.7.0,>=3.6.0->en-core-web-  
sm==3.6.0) (1.0.10)

Requirement already satisfied: cymem<2.1.0,>=2.0.2 in  
/usr/local/lib/python3.10/dist-packages (from spacy<3.7.0,>=3.6.0->en-core-web-  
sm==3.6.0) (2.0.8)

Requirement already satisfied: preshed<3.1.0,>=3.0.2 in

/usr/local/lib/python3.10/dist-packages (from spacy<3.7.0,>=3.6.0->en-core-web-sm==3.6.0) (3.0.9)  
 Requirement already satisfied: thinc<8.2.0,>=8.1.8 in  
 /usr/local/lib/python3.10/dist-packages (from spacy<3.7.0,>=3.6.0->en-core-web-sm==3.6.0) (8.1.12)  
 Requirement already satisfied: wasabi<1.2.0,>=0.9.1 in  
 /usr/local/lib/python3.10/dist-packages (from spacy<3.7.0,>=3.6.0->en-core-web-sm==3.6.0) (1.1.2)  
 Requirement already satisfied: srsly<3.0.0,>=2.4.3 in  
 /usr/local/lib/python3.10/dist-packages (from spacy<3.7.0,>=3.6.0->en-core-web-sm==3.6.0) (2.4.8)  
 Requirement already satisfied: catalogue<2.1.0,>=2.0.6 in  
 /usr/local/lib/python3.10/dist-packages (from spacy<3.7.0,>=3.6.0->en-core-web-sm==3.6.0) (2.0.10)  
 Requirement already satisfied: typer<0.10.0,>=0.3.0 in  
 /usr/local/lib/python3.10/dist-packages (from spacy<3.7.0,>=3.6.0->en-core-web-sm==3.6.0) (0.9.0)  
 Requirement already satisfied: pathy>=0.10.0 in /usr/local/lib/python3.10/dist-packages (from spacy<3.7.0,>=3.6.0->en-core-web-sm==3.6.0) (0.10.3)  
 Requirement already satisfied: smart-open<7.0.0,>=5.2.1 in  
 /usr/local/lib/python3.10/dist-packages (from spacy<3.7.0,>=3.6.0->en-core-web-sm==3.6.0) (6.4.0)  
 Requirement already satisfied: tqdm<5.0.0,>=4.38.0 in  
 /usr/local/lib/python3.10/dist-packages (from spacy<3.7.0,>=3.6.0->en-core-web-sm==3.6.0) (4.66.1)  
 Requirement already satisfied: numpy>=1.15.0 in /usr/local/lib/python3.10/dist-packages (from spacy<3.7.0,>=3.6.0->en-core-web-sm==3.6.0) (1.23.5)  
 Requirement already satisfied: requests<3.0.0,>=2.13.0 in  
 /usr/local/lib/python3.10/dist-packages (from spacy<3.7.0,>=3.6.0->en-core-web-sm==3.6.0) (2.31.0)  
 Requirement already satisfied: pydantic!=1.8,!1.8.1,<3.0.0,>=1.7.4 in  
 /usr/local/lib/python3.10/dist-packages (from spacy<3.7.0,>=3.6.0->en-core-web-sm==3.6.0) (1.10.13)  
 Requirement already satisfied: jinja2 in /usr/local/lib/python3.10/dist-packages (from spacy<3.7.0,>=3.6.0->en-core-web-sm==3.6.0) (3.1.2)  
 Requirement already satisfied: setuptools in /usr/local/lib/python3.10/dist-packages (from spacy<3.7.0,>=3.6.0->en-core-web-sm==3.6.0) (67.7.2)  
 Requirement already satisfied: packaging>=20.0 in  
 /usr/local/lib/python3.10/dist-packages (from spacy<3.7.0,>=3.6.0->en-core-web-sm==3.6.0) (23.2)  
 Requirement already satisfied: langcodes<4.0.0,>=3.2.0 in  
 /usr/local/lib/python3.10/dist-packages (from spacy<3.7.0,>=3.6.0->en-core-web-sm==3.6.0) (3.3.0)  
 Requirement already satisfied: typing-extensions>=4.2.0 in  
 /usr/local/lib/python3.10/dist-packages (from  
 pydantic!=1.8,!1.8.1,<3.0.0,>=1.7.4->spacy<3.7.0,>=3.6.0->en-core-web-sm==3.6.0) (4.5.0)  
 Requirement already satisfied: charset-normalizer<4,>=2 in



```

/usr/local/lib/python3.10/dist-packages (from
requests<3.0.0,>=2.13.0->spacy<3.7.0,>=3.6.0->en-core-web-sm==3.6.0) (3.3.2)
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-
packages (from requests<3.0.0,>=2.13.0->spacy<3.7.0,>=3.6.0->en-core-web-
sm==3.6.0) (3.6)
Requirement already satisfied: urllib3<3,>=1.21.1 in
/usr/local/lib/python3.10/dist-packages (from
requests<3.0.0,>=2.13.0->spacy<3.7.0,>=3.6.0->en-core-web-sm==3.6.0) (2.0.7)
Requirement already satisfied: certifi>=2017.4.17 in
/usr/local/lib/python3.10/dist-packages (from
requests<3.0.0,>=2.13.0->spacy<3.7.0,>=3.6.0->en-core-web-sm==3.6.0)
(2023.11.17)
Requirement already satisfied: blis<0.8.0,>=0.7.8 in
/usr/local/lib/python3.10/dist-packages (from
thinc<8.2.0,>=8.1.8->spacy<3.7.0,>=3.6.0->en-core-web-sm==3.6.0) (0.7.11)
Requirement already satisfied: confection<1.0.0,>=0.0.1 in
/usr/local/lib/python3.10/dist-packages (from
thinc<8.2.0,>=8.1.8->spacy<3.7.0,>=3.6.0->en-core-web-sm==3.6.0) (0.1.4)
Requirement already satisfied: click<9.0.0,>=7.1.1 in
/usr/local/lib/python3.10/dist-packages (from
typer<0.10.0,>=0.3.0->spacy<3.7.0,>=3.6.0->en-core-web-sm==3.6.0) (8.1.7)
Requirement already satisfied: MarkupSafe>=2.0 in
/usr/local/lib/python3.10/dist-packages (from jinja2->spacy<3.7.0,>=3.6.0->en-
core-web-sm==3.6.0) (2.1.3)

```

Download and installation successful

You can now load the package via `spacy.load('en_core_web_sm')`

```
[ ]: reviews_df['Date'] = pd.to_datetime(reviews_df['Date'])
reviews_df['Year'] = reviews_df['Date'].dt.year
```

Negative reviews over years

```
[ ]: # Filter the DataFrame
df_negative = reviews_df[reviews_df['Sentiment_Score'] < 0.2]
```

```
[ ]: import spacy
import string

nlp = spacy.load("en_core_web_sm")

exclude_words = set(['see', 'go', 'bad', 'know', 'story', 'film', 'titanic',
↳ 'movie', 'good', 'watch', 'ship', 'like',
↳ 'love', 'people', 'time', 'think', 'look', 'thing', 'want', 'make', 'mean', 'find', 'get', 'well', 'way'])

def clean_text(text):
    doc = nlp(text.lower())
    cleaned_text = []
```

```

    for token in doc:
        lemma = token.lemma_
        if lemma not in exclude_words and not token.is_stop and not token.
        ↳is_punct and not token.is_digit:
            cleaned_text.append(lemma)
    return " ".join(cleaned_text)
df_negative['Cleaned_Review'] = df_negative['Review'].apply(clean_text)

```

<ipython-input-33-ff1749b67e2b>:17: SettingWithCopyWarning:  
A value is trying to be set on a copy of a slice from a DataFrame.  
Try using .loc[row\_indexer,col\_indexer] = value instead

See the caveats in the documentation: [https://pandas.pydata.org/pandas-docs/stable/user\\_guide/indexing.html#returning-a-view-versus-a-copy](https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy)

```
df_negative['Cleaned_Review'] = df_negative['Review'].apply(clean_text)
```

```

[ ]: from collections import Counter
words = [word for review in df_negative['Cleaned_Review'] for word in review.
        ↳split()]
word_frequencies = Counter(words)
most_common_words = word_frequencies.most_common(20)
most_common_words

```

```

[ ]: [('cameron', 144),
      ('effect', 133),
      ('character', 115),
      ('sink', 97),
      ('hour', 94),
      ('special', 93),
      ('jack', 87),
      ('kate', 73),
      ('scene', 71),
      ('winslet', 69),
      ('dicaprio', 69),
      ('class', 65),
      ('rise', 64),
      ('real', 63),
      ('oscar', 62),
      ('acting', 61),
      ('plot', 60),
      ('life', 57),
      ('hate', 55),
      ('girl', 55)]

```

positive reviews over years

```
[ ]: df_positive = reviews_df[reviews_df['Sentiment_Score'] > 0.8]
df_positive['Cleaned_Review'] = df_positive['Review'].apply(clean_text)
```

<ipython-input-35-83d6a469f638>:2: SettingWithCopyWarning:  
A value is trying to be set on a copy of a slice from a DataFrame.  
Try using .loc[row\_indexer,col\_indexer] = value instead

See the caveats in the documentation: [https://pandas.pydata.org/pandas-docs/stable/user\\_guide/indexing.html#returning-a-view-versus-a-copy](https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy)  
df\_positive['Cleaned\_Review'] = df\_positive['Review'].apply(clean\_text)

```
[ ]: words = [word for review in df_positive['Cleaned_Review'] for word in review.
↳split()]
word_frequencies = Counter(words)
most_common_words = word_frequencies.most_common(20)
most_common_words
```

```
[ ]: [('cameron', 596),
      ('jack', 566),
      ('rise', 467),
      ('kate', 455),
      ('winslet', 435),
      ('dicaprio', 367),
      ('life', 336),
      ('character', 327),
      ('leonardo', 326),
      ('feel', 309),
      ('scene', 297),
      ('heart', 281),
      ('year', 256),
      ('romance', 253),
      ('effect', 249),
      ('amazing', 240),
      ('beautiful', 236),
      ('actor', 228),
      ('real', 204),
      ('end', 195)]
```

analyzing the top frequency words in 5-year segments

```
[ ]: for start_year in range(df_negative['Year'].min(), df_negative['Year'].max(),
↳5):
    end_year = start_year + 4
    segment_df = df_negative[(df_negative['Year'] >= start_year) &
↳(df_negative['Year'] <= end_year)]
    aggregated_tokens = [word for review in segment_df['Cleaned_Review'] for
↳word in review.split()]
```

```
top_words = Counter(aggregated_tokens).most_common(5)
print(f"Top words for {start_year}-{end_year}: {top_words}")
```

Top words for 1998-2002: [('effect', 92), ('character', 83), ('cameron', 82), ('special', 68), ('hour', 50)]

Top words for 2003-2007: [('cameron', 33), ('jack', 30), ('rise', 29), ('sink', 29), ('kate', 27)]

Top words for 2008-2012: [('action', 11), ('effect', 10), ('real', 10), ('awful', 10), ('sink', 9)]

Top words for 2013-2017: [('cameron', 11), ('scene', 7), ('awful', 4), ('kate', 4), ('long', 4)]

Top words for 2018-2022: [('jack', 20), ('life', 14), ('star', 14), ('sink', 12), ('rise', 11)]

```
[ ]: for start_year in range(df_positive['Year'].min(), df_positive['Year'].max(),
    ↪5):
    end_year = start_year + 4
    segment_df = df_positive[(df_positive['Year'] >= start_year) &
    ↪(df_positive['Year'] <= end_year)]
    aggregated_tokens = [word for review in segment_df['Cleaned_Review'] for
    ↪word in review.split()]
    top_words = Counter(aggregated_tokens).most_common(5)
    print(f"Top words for {start_year}-{end_year}: {top_words}")
```

Top words for 1998-2002: [('cameron', 134), ('winslet', 87), ('kate', 85), ('dicaprio', 72), ('jack', 65)]

Top words for 2003-2007: [('jack', 148), ('kate', 118), ('winslet', 115), ('cameron', 115), ('rise', 108)]

Top words for 2008-2012: [('jack', 114), ('cameron', 107), ('rise', 96), ('winslet', 71), ('kate', 65)]

Top words for 2013-2017: [('jack', 101), ('rise', 98), ('cameron', 59), ('kate', 58), ('character', 52)]

Top words for 2018-2022: [('cameron', 115), ('kate', 101), ('jack', 86), ('winslet', 86), ('leonardo', 79)]